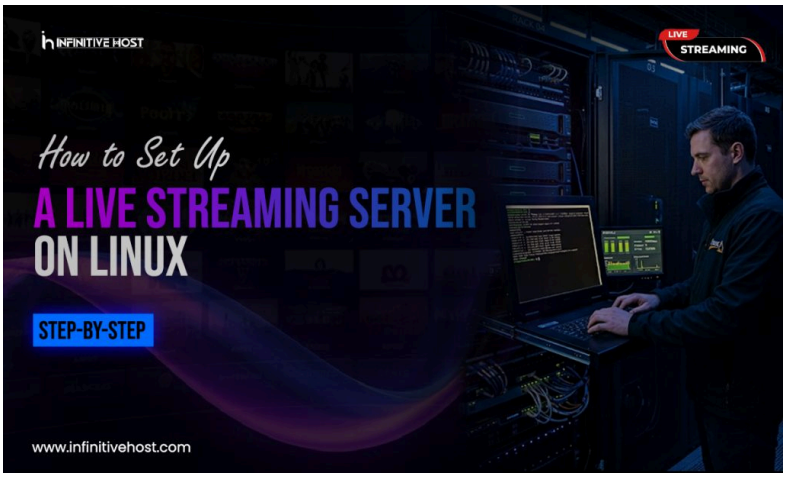

209 Views

How to Set Up a Live Streaming Server on Linux (Step-by-Step)

Live streaming's gone from a niche hobby to a top business tool. Whether you're gaming, hosting events, or just showing fun stuff, having your own streaming server really helps connect with viewers. It means full access—no annoying third-party limitations, no sudden shutdowns, and you save all the money.

This guide walks you through setting up a live streaming server on Linux, starting from scratch.

Why Run Your Own Streaming Server?

Running your own streaming server is better because popular platforms like Twitch cut your profits and limit what you can do. So, you're free to do whatever you want and keep more money too. When you host your own streaming server, you control latency, bitrate, storage, and audience access. For creators and businesses serious about quality, it's the only real option.

Your geographic choice matters too. A New York streaming server covers the US East Coast beautifully. Need broader reach? A Linux-based streaming server on the West Coast is your perfect bet for Pacific viewers. Serving both American and Canadian viewers at the same time? A Canadian datacenter with low cross-border latency always keeps buffering to a very little, and a powerful European-facing streaming infrastructure manages all transatlantic traffic without slowing down.

The reliable hosting service providers like InfinitiveHost always make this straightforward with InfinitiveHost dedicated streaming plans engineered mainly for media-powered tasks. They even provide streaming hosting with 25% off your first server, so getting started doesn't have to be costly.

What You Need Before Starting

- A Linux VPS or dedicated server (Ubuntu 20.04+ suggested)
- Root or sudo access
- A domain name (optional but very useful)
- General-level terminal knowledge
- Open ports: 1935 (RTMP), 8080 or 80 (HTTP)

Step 1: Update Your Server

Always start fresh:

```
sudo apt update && sudo apt upgrade -y
```

Step 2: Install Nginx with the RTMP Module

The most basic streaming server stack on Linux utilizes Nginx with the nginx-rtmp-module. RTMP (Real-Time Messaging Protocol) is the core of live video delivery.

sudo apt install nginx libnginx-mod-rtmp -y

Check out if it is installed or not:

nginx -v

Step 3: Set up Nginx for Live Streaming

Open the Nginx config file:

sudo nano /etc/nginx/nginx.conf

Include the following RTMP block at the bottom of the file (outside the http {} block):

```
rtmp {
  server {
    listen 1935;
    chunk_size 4096;

    application live {
      live on;
      record off;

      # Optional: push to multiple destinations
      # push rtmp://a.rtmp.youtube.com/live2/YOUR_KEY;
    }
  }
}
```

Save & close. Then restart Nginx:

sudo systemctl restart nginx

Your streaming server is now listening on port 1935.

Step 4: Include HLS Output (for Browser Playback)

RTMP alone won't play in today's browsers. Include HLS (HTTP Live Streaming) support inside your application live block:

```
application live {
  live on;
  record off;
  hls on;
  hls_path /var/www/html/hls;
  hls_fragment 3;
  hls_playlist_length 60;
}
```

Create the HLS directory:

sudo mkdir -p /var/www/html/hls

sudo chmod 777 /var/www/html/hls

Now include an HTTP block in the http {} section to serve the HLS files:

```
server {
  listen 8080;
  location /hls {
    types {
      application/vnd.apple.mpegurl m3u8;
      video/mp2t ts;
```

EN

```
    }
    root /var/www/html;
    add_header Cache-Control no-cache;
    add_header Access-Control-Allow-Origin *;
  }
}
```

Restart Nginx again:

```
sudo systemctl restart nginx
```

This always sets you up for a combined live and on-demand delivery setup — the audience can watch live or replay previously presented segments with the help of a playlist.

Step 5: Open Firewall Ports

```
sudo ufw allow 1935/tcp
sudo ufw allow 8080/tcp
sudo ufw reload
```

Step 6: Stream from OBS

In the case of OBS Studio:

1. Go to Settings → Stream
2. Set service to Custom
3. Enter your server URL: rtmp://your-server-ip/live
4. Set a stream key (e.g., mystream)
5. Click Start Streaming

Your streaming server is now live. Viewers can play your stream at:

```
http://your-server-ip:8080/hls/mystream.m3u8
```

Step 7: Enable Hardware-Accelerated Transcoding (Optional but Powerful)

If your chosen server has a GPU, setting up hardware-accelerated transcoding infrastructure significantly decreases CPU load. Install FFmpeg with hardware support:

```
sudo apt install ffmpeg -y
```

For NVIDIA GPUs, FFmpeg supports NVENC encoding. You can pipe your RTMP input through FFmpeg to transcode to multiple quality levels — 1080p, 720p, 480p — in real time without melting your CPU.

This is where providers like infinitive host stand out. Their InfinitiveHost dedicated streaming plans include GPU-capable instances that are pre-configured for exactly this kind of hardware-accelerated transcoding infrastructure, saving hours of setup.

Step 8: Secure Your Stream

Don't leave your streaming server open. Add a stream key check in Nginx:

```
application live {
  live on;
  record off;
  on_publish http://localhost/auth;
}
```

Or simply keep your stream key private and rotate it regularly.

Conclusion

Setting up a live streaming server on Linux isn't as complicated as it looks. With Nginx, RTMP, and HLS configured correctly, you get a production-ready broadcast setup that you fully own. Whether you choose a New York streaming server for East Coast audiences, a Linux-optimized streaming server on the West Coast, or a Canadian datacenter with low cross-border latency, the fundamentals covered here work across all of them.

For those who want a hassle-free start, InfinitiveHost dedicated streaming plans come ready-to-go with the infrastructure already optimized — and with streaming hosting with 25% off your first server, it's worth a look before you spin up everything manually.

Read Related – Complete Guide to Streaming Servers & Live Streaming VOD (2026)

EN

FAQs

- What's the best Linux distro for a streaming server?
- How much bandwidth does a live streaming server need?
- Can I run a streaming server on a cheap VPS?
- What's the difference between RTMP and HLS?
- Is a combined live and on-demand delivery setup hard to maintain?

Related Blogs

When Your Live Stream Keeps Buffering (And How...

infi_admin | July 4, 2026

When Your Live Stream Keeps Buffering (And How to Fix It With

Leave a Reply

Your email address will not be published. Required fields are marked *

Name *

Email *

Website

Comment *

Save my name, email, and website in this browser for the next time I comment.

Post Comment

EN

PREV
Bare Metal GPU Server vs Virtual GPU Server:...

NEXT
GPU Server for Video Generation AI: Sora Alternatives,...

INFINITIVE HOST

Our Address: Office No: 201-202, Second Floor, Elements Mall, DCM Ajmer Road, Jaipur, 302021, Rajasthan, India

US: NO .1910, Thomes Avenue, Cheyenne, Wyoming. 82001

Around the Web

An ISO 27001 & 9001 Certified Company

Platforms

- AWS
- Google Cloud
- Azure
- Kubernetes
- Odoo
- Vtiger Solutions

Company

- About Us
- Our Team
- Work Culture
- Awards & Recognitions
- Certificates
- Payment Options
- FAQs

Digital Solutions

- Web Development
- App Development
- Digital Marketing

Solutions

- Server Management
- Backup Solutions

Email & Security

- Office 365
- Zimbra
- SSL
- Cyber Security

© 2026 Infinitive Host. All rights reserved

EN